Low Complexity Scalable Iterative Algorithms for IEEE 802.11p Receivers

Olivier Goubet, Gwilherm Baudic, Frédéric Gabry *Student Member, IEEE,* and Tobias J. Oechtering, *Senior, IEEE*

***Abstract*—In this paper, we investigate receivers for Vehicular to Vehicular (V2V) and Vehicular to Infrastructure (V2I) communications. Vehicular channels are characterized by multiple paths and time variations, which introduces challenges in the design of receivers. We propose an algorithm for IEEE 802.11p compliant receivers, based on Orthogonal Frequency Division Multiplexing (OFDM). We employ iterative structures in the receiver as a way to estimate the channel despite variations within a frame. The channel estimator is based on factor graphs, which allow the design of soft iterative receivers while keeping an acceptable computational complexity. Throughout this work, we focus on designing a receiver offering a good complexity performance trade-off. Moreover, we propose a scalable algorithm in order to be able to tune the trade-off depending on the channel conditions. Our algorithm allows reliable communications while offering a considerable decrease in computational complexity. In particular, numerical results show the trade-off between complexity and performance measured in computational time and BER as well as FER achieved by various interpolation lengths used by the estimator which both outperform by decades the standard least square solution. Furthermore our adaptive algorithm shows a considerable improvement in terms of computational time and complexity against state of the art and classical receptors whilst showing acceptable BER and FER performance.**

I. Introduction

INTELLIGENT Transportation Systems (ITS) are expected to dramatically improve safety on the roads, by offering applications such as collision-avoidance at intersections and fast spreading of emergency warnings. Additional scenarios and applications are examined in [1]. Improved traffic management [2], reduced impact on the environment and enhanced comfort for the driver and passengers are also expected. Such applications require not only communications between vehicles and infrastructures situated along the roads (designated by Vehicle-to-Infrastructure communications and noted V2I), but also Vehicle-to-Vehicle (V2V) communications and the possibility to create ad hoc networks. Moreover, applications such as the aforementioned detection of vehicles approaching an intersection rely on the ability to communicate beyond the Line-of-Sight (LOS), which is generally not possible with current systems. Wireless communications for both V2V and V2I are thus the enabling technology for developing ITS. An international standard called Wireless Access for Vehicular Environment (WAVE) has been developed specifically for this purpose. It specifies that the physical layer (PHY) is based on IEEE 802.11p, which is included in the IEEE 802.11 standard, widely used in Wireless Local Access Networks (WLAN).

As highlighted during measurement campaigns, such as the one described in [3], vehicular channels are characterized by multipath propagation and high Doppler-shifts, leading to doubly selective channels, varying over one frame. Thus, wireless communications in vehicular environments are challenging, especially in the case of V2V, where the transmitter, receiver, as well as reflectors are in motion. Note that multipath channels, while increasing the difficulty of communicating with low error rates, are also beneficial when applications rely heavily on communicating in Non-Line-of-Sight (NLOS) situations. A thorough characterization of the vehicular channel in various types of environment is given in [1]. The IEEE 802.11p standard uses Orthogonal Frequency Division Multiplexing (OFDM) modulation, which alleviates the issue of InterSymbol Interference (ISI). However, the pilot pattern concentrates most of the information in the first two symbols, the density of pilots in the subsequent frame being kept low. Combined with the necessity to perform frame-by-frame decoding [4], the channel estimation becomes challenging. Many receivers have been designed using channel estimation methods classically present in OFDM systems, such as the ones described in [5]–[10], where channel estimation is performed either in the time domain, assuming a constant channel over a frame, or in the frequency domain, using interpolation methods. The results obtained with such receivers highlight some limitations in the standard, as explained in [11]. In particular, the pilot spacing in frequency is not sufficient, as the coherence bandwidth is generally too small. Consequently, time domain estimation is usually preferred [8], even if coherence time causes a similar problem when long frames are transmitted. Two approaches to tackle this issue can be distinguished. On the one hand, attempts to modify the standard in order to make it more suitable to the environment have been proposed. While in [12] the authors suggest introducing a midamble containing pilot symbols, allowing a more accurate tracking of the channel variations, it is proposed in [13] to add information in a

The work was supported in part by the Strategic Research Agenda Program, Information and Communication Technology – The Next Generation (SRA ICT – TNG), through the Swedish Government and the EIT ICT labs in tasks IMS 13066-A1304 and A1305.

postamble. In [14] spectral keying is introduced into conventional OFDM present in IEEE 802.11p. Finally, the effect of using a larger cyclic prefix is investigated in [15]. On the other hand, it is possible to develop receivers fully compliant with IEEE 802.11p, using advanced channel estimation techniques to counteract the effects of the channel.

In [3] the use of iterative receivers is suggested, performing Joint Channel Estimation and Decoding (JCED), as a way to track the variations of the channel within a single frame, which is an inevitable phenomenon at higher speed or when longer frames are transmitted. The concept behind iterative receivers is described in [16], where estimates of the coded bits are fed back to the channel estimator and used as pilots, as a way to refine the estimation of the channel. In [4], [13], a receiver based on this principle is proposed, and reaches good performance in terms of Bit Error Rate (BER) and Frame Error Rate (FER). In order to increase the performance, the authors in [3] use a 1x2 Single Input Multiple Output (SIMO) system in addition to the iterative structure. However, implementing these methods comes at the cost of a higher complexity. In [17], [18], complexity efficient receivers were investigated. As expected, the performance decreases, highlighting the need to design receivers with a good performance complexity trade-off as done in this work.

A key factor in the design of iterative receivers is the ability to access soft feedback information. Iterative receivers propagating exclusively soft information can be designed using the framework provided by factor graphs [19]. Based on previous work presented in [20], [21], a graph-based receiver for Multiple Input Multiple Output (MIMO) OFDM is proposed in [22], [23]. In these works, a message passing algorithm is applied on a three dimensional graph representing the channel. Performance over doubly selective channels is promising and the complexity remains relatively low.

In this paper, we aim at designing receivers compliant to the IEEE 802.11p standard and allowing vehicle to vehicle communications. We focus on achieving good performance in terms of error rate while maintaining a reasonable computational complexity so as to make the design suitable for a later hardware implementation.

To implement the channel estimator, which is the crucial part of the receiver, we propose to improve and adapt to the standard the framework developed in [22]. First, the receiver must be adapted to the pilot pattern structure. Secondly, we take advantage of this structure to improve the channel estimator. Indeed, while most receivers are model based, we introduce a solution that estimates the channel statistical properties based on the received frame, instead of relying on a channel model. This leads to a very robust design universally good for various vehicular channels.

The main contributions of this work are summarized as follows:

- We adapt the framework developed in [22] to make it suitable to the IEEE 802.11p standard. The channel estimator takes into account the pilot pattern. Suitable decoding, demapping and deinterleaving are implemented.
- We further improve the channel estimator by taking advantage of the pilot pattern in order to avoid any dependency on channel models. Channel statistical properties are now estimated based on the received signal.
- An adaptive structure is proposed as a way to scale the trade-off between performance and complexity. This also enables the design of a flexible receiver able to adapt to various environments and qualities of service.
- We validate the design through a theoretical analysis of the complexity and link level simulations.
- We present measures of the BER, the FER, and the complexity, which shows that the trade-off between complexity and performance achieved by our receiver is better than the one reached by other receivers found in the literature.

TABLE I
OFDM PARAMETERS IN IEEE 802.11P

Parameter	Notation	Value
Number of OFDM subcarriers	N_{sc}	64
Number of data subcarriers	N_{scd}	48
Number of used subcarriers	N_{scu}	52
Number of samples in the cyclic prefix	G	16
Number of OFDM symbols	N_s	variable
Channel spacing	B	10 MHz
Sampling time	T_s	100 ns
Symbol period	T_{OFDM}	8 µs

This paper is organized as follows: The signal and channel models are presented in Section II. In Section III, we describe the system model. A theoretical analysis of the complexity as well as a comparison with other state-of-the-art receivers are provided in Section IV. The numerical results of the link level simulations are shown and discussed in Section V. Finally, Section VI concludes this paper.

Notation: We denote by $p_Y(y)$ the probability density function (pdf) of a random variable Y, and by $\mathbb{E}\{Y\}$ its expected value. $\Pr(A)$ denotes the probability of an event A. z^* is the complex conjugate of $z \in \mathbb{C}$. Finally, if $a \in \mathbb{R}$, then $\lfloor a \rfloor$ is the largest integer lower than or equal to a.

II. SIGNAL AND CHANNEL MODEL

In this section, we first describe the signal model. The transmitted signals are generated by an IEEE 802.11p compliant transmitter. A brief presentation of the standard relevant parts is provided here. We then introduce a channel model that will be used to simulate the system, and deduce a model for the received signal.

A. Transmitted Signal Model

The PHY layer is based on the IEEE 802.11p standard, based itself on 802.11 OFDM.

The OFDM parameters are given in Table I. The total bandwidth B of the signal is divided into N_{sc} OFDM subcarriers. A guard interval of length G is appended before each symbol; this interval contains a cyclic prefix, that is, a repetition of the last samples of the symbol, in order to mitigate ISI. The sampling time is $T_s = 1/B$ and the symbol period is $T_{OFDM} = (N_{sc} + G)/B$. The total number of symbols

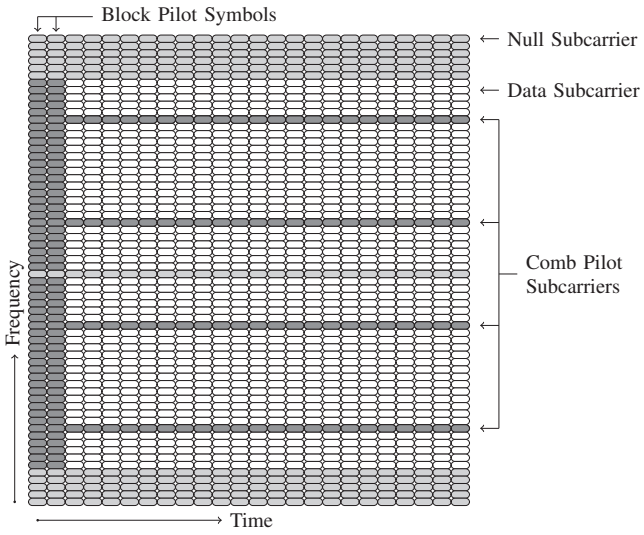


Fig. 1. Frame structure in an IEEE 802.11p system.

per frame, denoted by N_s, varies depending on the desired payload.

The standard also defines the repartition of pilot symbols in the OFDM frame, as shown in Fig. 1. Eleven outer carriers are set to null to avoid frequency leakage to the other frequency bands. The DC (Direct Current) subcarrier is also null to avoid undesirable DC offset [24]. Four subcarriers, denoted as *comb pilot* subcarriers, are reserved for pilot symbols. Thus forty-eight carriers remain to carry data. For every non-null subcarrier, the first two symbols are reserved for pilots; these symbols are denoted *block pilots*. This structure mixing both types of pilots is called *block-comb*, and typically offers good performance in the case of quasi-static channels. In this case, block pilots are usually used for channel coefficient estimation and comb pilots for frequency offset detection and compensation.

Estimating the channel using only the information from the block pilots requires that the frame duration is much shorter than the coherence time. In [25], the case of a high speed scenario is presented. Considering a relative speed $v = 85\,\mathrm{m\,s^{-1}}$ ($306\,\mathrm{km\,h^{-1}}$), $f_c = 5.9\,\mathrm{GHz}$, and $c = 3 \times 10^8 \mathrm{m\,s^{-1}}$, the maximum Doppler spread, given by $f_d = v f_c / c$, is around 1700 Hz, meaning that the channel will significantly change in $1/f_d \approx 600\,\mu\mathrm{s}$, which corresponds to about 75 OFDM symbols. When the rate is 6 Mbps, it corresponds to a payload of 450 bytes. Frame lengths are typically in the range 300-800 bytes, and thus channel estimation solely based on block pilots will not always be sufficient. Note however that Intercarrier Iterference (ICI) should not be a problem, as $f_d T_{OFDM} \ll 1$. It is also possible to perform channel estimation in the frequency domain, using the information from the comb pilots, and obtaining the other channel coefficients by applying interpolation methods. In this case, we are interested in the coherence bandwidth:

$$B_c = \frac{1}{T_{ch}}. \tag{1}$$

Considering the carrier spacing and the pilot allocation, the delay spread of the channel T_{ch} should be lower than $450\,\mathrm{ns}$ to ensure proper channel estimation based on comb pilots. Most delay spreads that were measured were actually longer [1]. Thus, channel estimation should be carefully designed, taking advantage of both the block and comb pilots, in order to estimate the channel sufficiently well. As we will show in Section III, iterative structures can compensate for the low density of pilots.

The symbols transmitted in the OFDM frame are drawn from four possible constellations, namely Binary Phase Shift Keying (BPSK), Quadrature Phase Shift Keying (QPSK), 16 Quadrature Amplitude Modulation (QAM) and 64-QAM. Before being mapped to the symbols, the bit stream goes through a convolutional encoder of rate ½, generated by the polynomials $g_1 = (133)_8$ and $g_2 = (171)_8$. Puncturing can be applied to increase the coding rate. The coded bits are interleaved twice to allow for a more uniform distribution of errors and to avoid long runs of low reliability [26]. The various possible combinations of mappings and coding rates lead to transmission rates in the range 3-27 Mbps.

B. Channel Model and Received Signal

We consider a time-varying multipath channel model. Such a channel can be represented by a tapped delay line, where each tap has an amplitude $\alpha_i(t)$ which varies around a predefined average value, a delay $\tau_i(t)$, and is affected by a Doppler shift $f_{d,i}$. A channel's impulse response consisting of L_{ch} paths can thus be expressed as:

$$h_t(t,\tau) = \sum_{i=1}^{L_{ch}} \alpha_i(t)\delta(\tau - \tau_i(t))e^{j2\pi f_{d,i}t}. \tag{2}$$

The average power of the taps can be chosen according to a statistical model, for example exponentially decaying or uniform as in [13] or following an empirical model based on measurements, as in [27]. The Doppler shifts are chosen according to a distribution such as the Jakes spectrum or a flat spectrum, in $[-f_{d,max}, f_{d,max}]$, where $f_{d,max} = v f_c / c$. We define the normalized Doppler shift as $\nu_i = f_{d,i} T_s$. This model accounts for the double selectivity of the channel.

We consider delay spreads shorter than the cyclic prefix, which allows the system to alleviate completely the ISI between OFDM symbols. Each transmitted symbol in the time-frequency space is thus affected by a single channel coefficient [28, Chapter 8]. If the delay spread is actually larger than the cyclic prefix, then ISI will not be completely removed, and the performance will be affected. After OFDM demodulation, the received signal can be expressed in the frequency domain as

$$y[k,l] = h[k,l] \cdot x[k,l] + z[k,l], \tag{3}$$

for $k = 1, \ldots, N_{sc}$ and $l = 1, \ldots, N_s$ and where $x[k,l]$ is the transmitted symbol on the k^{th} subcarrier and at the l^{th} symbol index, $h[k,l]$ and $z[k,l]$ respectively the channel coefficient and noise sample affecting this symbol, and $y[k,l]$ the received signal [22]. We refer the reader to [29] for a detailed derivation of the relation between $h_t(t,\tau)$ and $h[k,l]$.

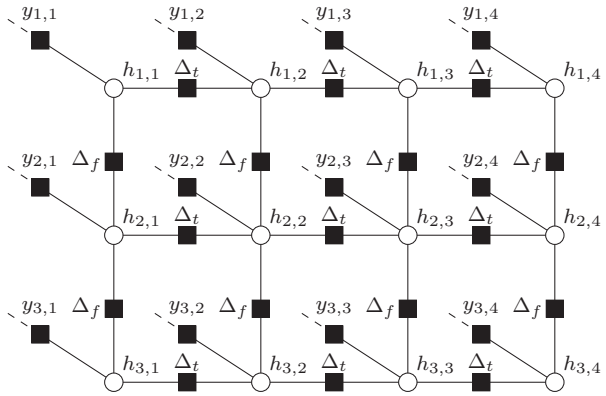


Fig. 2. Exchange of messages at the coefficient nodes.

The noise samples are considered to be complex Gaussian distributed, independent and identically distributed:

$$z[k,l] \sim \mathcal{CN}(0, \sigma_z^2). \tag{4}$$

III. System Description

In this section, we describe the algorithm implemented in the receiver. We propose to adapt and improve the framework developed in [20]–[23], based on factor graphs. These are bipartite graphs where nodes can exchange messages. Factor graphs applied to digital communications, with a focus on receiver design, were introduced in [19]. They allow to create soft receivers with low complexity. Frames are decoded by applying a message passing algorithm on the graph representing the receiver, as explained in [30]. After presenting the framework, we introduce the proposed model, designed to make the algorithm suitable for vehicular communications and take advantage of the features of the IEEE 802.11p standard. Moreover, we focus on the trade-off between complexity and performance, and propose a scalable algorithm allowing to tune this trade-off.

A. Graph-Based Soft Iterative Channel Estimator

In [20]–[23], a soft channel estimator based on factor graphs was proposed. It offered good performance in terms of BER, while keeping a relatively low computational complexity. In this section, we present this solution.

1) Factor Graph: Fig. 2 shows the factor graph describing the channel estimator, in the case of a simplified example with a frame consisting of three subcarriers and four OFDM symbols. Note that $y[k,l]$ and $h[k,l]$ are denoted as $y_{k,l}$ and $h_{k,l}$ for a better readability. We distinguish three different types of nodes:

- the coefficient nodes representing the channel coefficients and denoted as $h_{k,l}$,
- the transfer nodes Δ_t and Δ_f, connecting the channel coefficients,
- the observation nodes $y_{k,l}$ corresponding to the received signal.

Edges represented with dashed lines correspond to the connections to the decoder. To avoid short cycles, only the transitions involving shifting by one coefficient in either time or frequency (not both) are considered, leading to the present structure. Note however that cycles are still present and inevitable; the scheduling has to be chosen carefully, so that the exchange of intrinsic information in a cycle is low. In other words, a node must have as little dependence as possible on its input messages. Complete independence is achievable only when no cycles are present. In the following, we drop the indices in the notation when it is not ambiguous.

2) Observation Nodes: In an observation node, the likelihood of the observation y given a certain channel coefficient, namely $p(y|h)$, is computed according to:

$$p(y|h) = \sum_{x \in \mathcal{X}} \Pr(x) p(y|h,x) \tag{5}$$

$$= \sum_{x \in \mathcal{X}} \Pr(x) \frac{1}{\pi \sigma_z^2} \cdot \exp\left(-\frac{|hx-y|^2}{\sigma_z^2} \right), \tag{6}$$

where $\mathcal{X}$ designates the symbol alphabet. $p(y|h)$ is thus a Gaussian mixture. We can assume that $p(y|h)$ is Gaussian if there exists a symbol x_i such that $\Pr(x_i) \gg \Pr(x_j)$, for all $j \neq i$. This hypothesis is valid if the symbols are detected with a large enough reliability.

To reduce the complexity, we assume, as suggested in [31, Eq. 22–23], that

$$p(h) \propto p(y|h), \tag{7}$$

and the distribution of the channel coefficients h can be approximated to be Gaussian:

$$h \sim \mathcal{CN}(\mu_h, \sigma_h^2). \tag{8}$$

Note that with this approach we do not assume that the priors are known at the receivers since they are approximated and therefore our algorithm is stricly speaking non-Bayesian, even if it uses prior probabilities about the channel, which is usually a Bayesian approach. This approximation makes the proposed design universal and not a design which works well for a specific channel model. Using this Gaussian approximation, it is possible to compute the mean and variance of h as:

$$\mu_h = Cy \sum_{x \in \mathcal{X}} \frac{\Pr(x)}{x|x|^2} \tag{9}$$

and

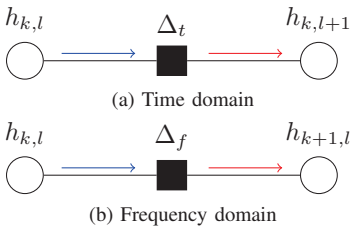
$$\sigma_h^2 = C(\sigma_z^2 + |y|^2) \sum_{x \in \mathcal{X}} \frac{\Pr(x)}{|x|^4} - \mu_h, \tag{10}$$

where the normalization factor C is given by

$$C = \frac{1}{\sum_{x \in \mathcal{X}} \Pr(x)/|x|^2}. \tag{11}$$

μ_h represents the hard estimate of h and σ_h^2 is a measure of the reliability of the hard estimate.

3) Transfer Nodes: The exchange of information at the transfer nodes is shown in Fig. 3a and Fig. 3b for the time domain and the frequency domain respectively. The transfer nodes Δ_t and Δ_f account for the variations of the channel respectively in time and frequency. In particular, the transfer nodes account for movement features, such as the Doppler effect. More details on these nodes are given in Section III-B.



(a) Time domain

(b) Frequency domain

Fig. 3. Exchange of messages in the transfer nodes. At a certain time, a message will flow only in one direction. Considering the entire estimation, all messages are going in both directions as described in Figure 6. For better readability, only one way propagation is shown in the Figure.

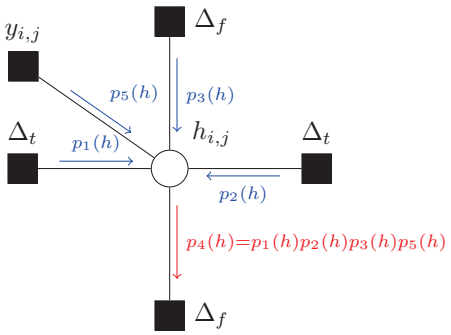


Fig. 4. Exchange of messages at the coefficient nodes.

4) Coefficients Nodes: The propagation of the messages at the coefficient nodes is depicted in Fig. 4. When computing the outcoming message on edge i, we consider all edges $j \neq i$, to ensure that only extrinsic information is transmitted. The outcoming messages are thus computed as:

$$p_i(h) = \left(\prod_{j=1, j\neq i}^{N} p_j(h) \right). \tag{12}$$

$p_i(h)$ is a Gaussian distribution, with mean μ_i and variance σ_i^2, obtained as:

$$\mu_i = \frac{\sum_{j=1,j\neq i}^{N} \frac{\mu_j}{\sigma_j^2}}{\sum_{j=1,j\neq i}^{N} \frac{1}{\sigma_j^2}} \quad \text{and} \tag{13}$$

$$\sigma_i^2 = \frac{1}{\sum_{j=1,j\neq i}^{N} \frac{1}{\sigma_j^2}}. \tag{14}$$

After running the Sum Product Algorithm (SPA) on the factor graph, the messages $p_{h\to obs}(h)$ (e.g., $p_5(h)$ in Fig. 4), characterized by μ_h and σ_h^2, are available and used as the channel estimates. Note that we perform indeed soft channel estimation, as each channel coefficient estimate conveys a measure of reliability, contained in σ_h^2.

B. Proposed Algorithm for an Iterative Receiver in the Case of IEEE 802.11p

In [5], it was pointed out that model based channel estimators may behave poorly in real environments. We propose a new way of deriving the transfer nodes, which contain the information relative to the channel statistical properties, by taking advantage of the structure of the pilot pattern. Also, the non-symmetric pilot pattern forces us to adapt the scheduling. We then propose a structure for the remaining of the factor graph, allowing to detect frames in the case of IEEE 802.11p compliant systems. Finally, we introduce a scalable algorithm.

1) Approximation of the Transfer Nodes: In general, Δ is given by:

$$\Delta[k,k',l,l'] = h[k,l] - h[k',l']. \tag{15}$$

Under a wide sense stationary hypothesis, which can be a reasonable assumption in certain situations [32], Equation (15) becomes:

$$\Delta[k',l'] = h[k,l] - h[k+k',l+l']. \tag{16}$$

$\Delta[k,k',l,l']$ is Gaussian distributed with zero mean and variance $\sigma_\Delta^2[k,k',l,l']$:

$$\Delta[k,k',l,l'] \sim \mathcal{N}(0, \sigma_\Delta^2[k,k',l,l']). \tag{17}$$

The mean is propagated along the graph, while the uncertainty increases at each transfer node:

$$\mu_h[k',l'] = \mu_h[k,l] \quad \text{and} \tag{18}$$

$$\sigma_h^2[k',l'] = \sigma_h^2[k,l] + \sigma_\Delta^2[k,k',l,l']. \tag{19}$$

The degree of increase of the uncertainty, defined by σ_Δ^2, depends on the correlation properties of the channel:

$$\begin{aligned}\sigma_\Delta^2[k,k',l,l'] &= \mathbb{E}\{|h[k,l] - h[k',l']|^2\} \\ &= \mathbb{E}\{|h[k,l]|^2\} + \mathbb{E}\{|h[k',l']|^2\} \\ &\quad - 2\mathrm{Re}[\mathbb{E}\{h^*[k,l]h[k',l']\}].\end{aligned} \tag{20}$$

In [22], isolated pilots are regularly inserted in the frame, allowing for a good tracking of the channel, but making it difficult to determine the channel's statistical properties, solely based on the received signal. This difficulty could possibly be alleviated, e.g. by considering a parameterized model for the autocorrelation function of the channel. However, the IEEE 802.11p standard's pilot pattern includes adjacent pilots, both spread in time (comb pilots) and frequency (block pilots). This gives the possibility to compute estimates of the channel's correlation, required to determine the transfer nodes. Let $\hat{r}_{h,f}(k,k')$ be an estimate of $\mathbb{E}\{h[k,l]h^*[k',l]\}$ and $\hat{r}_{h,t}(l,l')$ an estimate of $\mathbb{E}\{h[k,l]h^*[k,l']\}$. Computing $\hat{r}_{h,f}(k,k')$ and $\hat{r}_{h,t}(l,l')$ requires the estimates $\hat{h}_p$ of the true channel coefficients h_p, obtained using the pilot symbols x_p:

$$\hat{h}_p = \frac{y_p}{x_p} = h_p + \frac{z}{x_p}. \tag{21}$$

We then have:

$$\hat{r}_{h,f}(k,k') = \frac{1}{|\mathcal{L}_p|} \sum_{l\in\mathcal{L}_p} \hat{h}[k+k',l]\hat{h}^*[k',l] \quad \text{and} \tag{22}$$

$$\hat{r}_{h,t}(l,l') = \frac{1}{|\mathcal{K}_p|} \sum_{k\in\mathcal{K}_p} \hat{h}[k,l+l']\hat{h}^*[k,l'], \tag{23}$$

where $\mathcal{L}_p$ designates the set of indices of the block pilots and $\mathcal{K}_p$ the set of indices of the comb pilots. Note that since the number of non-zero terms in the above sums depend on the

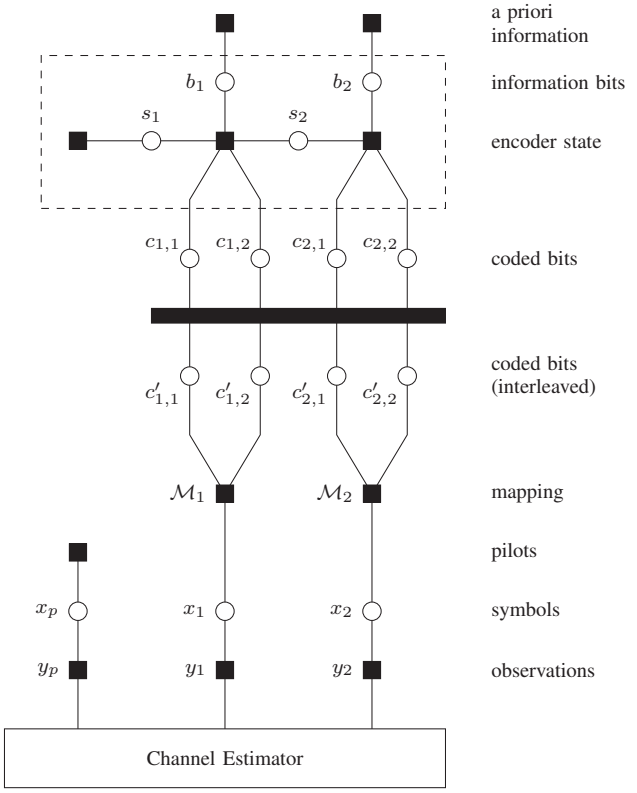


Fig. 5. Complete factor graph of the receiver.

indices k' and l', the estimates $\hat{r}_{h,f}(k,k')$ and $\hat{r}_{h,t}(l,l')$ are biased.

Under a wide sense stationary hypothesis, Equations (22) and (23) become:

$$\hat{r}_{h,f}(k) = \frac{1}{N_{scu}|\mathcal{L}_p|} \sum_{l\in\mathcal{L}_p} \sum_{k'=0}^{N_{scu}-k-1} \hat{h}[k'+k,l]\hat{h}^*[k',l] \quad \text{and} \tag{24}$$

$$\hat{r}_{h,t}(l) = \frac{1}{N_s|\mathcal{K}_p|} \sum_{k\in\mathcal{K}_p} \sum_{l'=0}^{N_s-l-1} \hat{h}[k,l'+l]\hat{h}^*[k,l']. \tag{25}$$

Because of the structure of the factor graph, we only need the correlation between two consecutive symbols in time or frequency. The correlation properties in time are computed using the comb pilots, while the correlation properties in frequency are computed using the block pilots.

If the channel is considered wide sense stationary in time over a frame, $\hat{r}_h(l)$ is estimated using all the pilots, and Δ_t is constant. If not, Δ_t is time varying. However, depending on the channel properties, Δ_t can be considered constant over only a certain time span. The shorter the span, the fewer pilots are used to compute a single transfer node. A similar discussion can be made regarding Δ_f.

2) Computation of Log-Likelihood Ratios (LLRs) and Decoding: Fig. 5 shows the complete factor graph of the receiver, in the case of a simple example. The channel estimator is as presented in Section III-A. The upwards messages coming out of the observation nodes are the likelihoods regarding the

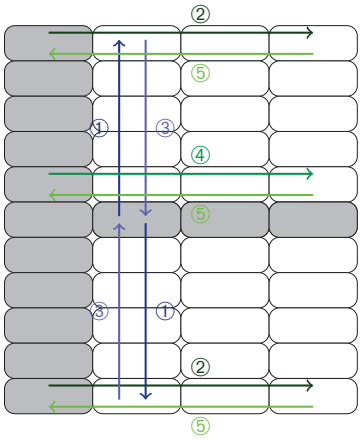


Fig. 6. Scheduling of the message passing algorithm.

symbols x:

$$\begin{aligned} p(y|x) &= \int p(y|h,x)p(h)dh \\ &= \frac{1}{\pi(\sigma^2|x|^2+\sigma_z^2)} \cdot \exp\left(-\frac{|y-\mu x|^2}{\sigma^2|x|^2+\sigma_z^2}\right). \end{aligned} \tag{26}$$

The mapping nodes compute the Log Likelihood Ratios (LLRs) of the coded bits:

$$\begin{aligned} LLR(c) &= \sum_{x\in\mathcal{X}_0} \log(p(y|x)) - \sum_{x\in\mathcal{X}_1} \log(p(y|x)) \\ &= \sum_{x\in\mathcal{X}_0} \left(-\frac{|y-\mu x|^2}{\sigma^2|x|^2+\sigma_z^2}\right) \\ &\quad - \sum_{x\in\mathcal{X}_1} \left(-\frac{|y-\mu x|^2}{\sigma^2|x|^2+\sigma_z^2}\right), \end{aligned} \tag{27}$$

where $\mathcal{X}_0$ (resp. $\mathcal{X}_1$) designates the set of symbols such that $c=0$ (resp. $c=1$). The LLRs are then de-interleaved, before entering the decoder. The BCJR algorithm, introduced in [33], is used to obtain the LLRs of the information bits from the LLRs of the coded bits and the a priori probabilities of the information bits, which are fed to the decoder. The algorithm also gives an updated version of the LLRs of the coded bits used to determine the extrinsic information. An implementation of the algorithm is suggested in [34, Chapter 4]. The description of the BCJR with factor graphs is given in [35, Chapter 8]. After the end of the iterative process, a hard decision is made on the coded bits based on the following decision rule:

$$\hat{b} = \begin{cases} 0 & \text{if } LLR(b) > 0 \\ 1 & \text{if } LLR(b) < 0. \end{cases} \tag{28}$$

3) Scheduling: As mentioned in Section III-A, the factor graph describing the channel estimator has cycles. The scheduling must be chosen to minimize their effect. A solution is to proceed dimension by dimension, i.e. propagating messages back and forth in one direction (vertical for instance), and then perform a similar task in the other direction (horizontal in our example). This also allows a pipeline implementation, thus reducing the complexity. In order to avoid computing

messages which do not carry information, it is important that the passing of a message begins either at the position of a pilot symbol, or at a position where messages have already been computed in the other direction. Due to the sequential approach of the factor graph algorithm, we need to carefully define the sequential order of the estimator which is defined by the proposed scheduling strategy. This strategy is based on an heuristic design exploiting the pilot symbols in the frame as efficiently as possible and trying to avoid short cycles to limit the amplification of errors. Furthermore, it easily allows us to introduce the interpolation approach from Section III-B4.

The proposed scheduling in the channel estimator is represented on Fig. 6, which shows the time frequency spreading of the nodes. For simplicity, Fig. 6 only shows a small part of the OFDM frame represented in Fig. 1. In order to not clutter Fig. 6, we do not draw arrows for every subcarrier or data symbol propagation. For instance, if we look at step 4, only one subcarrier propagation is represented by a green arrow, while the forward propagation is actually carried on all remaining subcarriers, as explained in the following steps. The procedure is described in chronological order as follows:

① forward propagation of the messages in the frequency domain, as it is typically the one with the fastest variations, from a pilot subcarrier to the edge of the frame and for all data symbols;
② forward propagation of messages in the time domain for the two outer subcarriers;
③ backward phase in the frequency domain for all data symbols;
④ forward propagation in the time domain on the remaining subcarriers ;
⑤ backward propagation in the time domain on all subcarriers;
⑥ computation of the messages coming from the coefficient nodes to the observation nodes.

Messages are computed according to Equations (13), (14), (18), and (19). LLRs are then computed, followed by the deinterleaving step and the BCJR decoding, providing both the LLRs for the information bits and the coded bits. These are used to compute extrinsic information, which is re-interleaved and fed to the channel estimator to compute new channel estimates through the mapper and the observation nodes.

4) Scalable Algorithm: In this work, we are interested in obtaining a good complexity performance trade-off. However, the quality of this trade-off depends on several conditions, such as the SNR (Signal-to-Noise Ratio), the frame length or the relative speed of the vehicles. We propose here a method to adjust the complexity of the algorithm. This is expected to have an impact on the performance as well, and ultimately on the complexity performance trade-off.

The underlying principle is to estimate only a subset of the channel coefficients. We designate by L the adaption factor in the time domain and decimate the frame by this factor. Note that both pilots at the beginning of the frame are kept, and only the data symbols are affected by the decimation. The new factor graph describing the channel estimator is shown on Fig. 7, for subcarrier k. We denote by N_p the number of block pilots and define $M = \lfloor (N_s + N_p - 3)/L \rfloor$. The coefficients

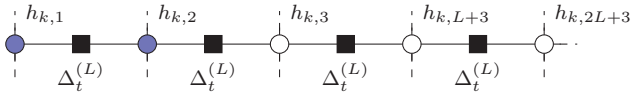


Fig. 7. Factor graph of the channel estimator after adaptation.

$h_{k,mL+N_p+1}$, with $m \in \{1, \ldots, M\}$, are estimated. The other channel coefficients are obtained by piecewise constant interpolation, as it has the lowest computational complexity:

$$h_{k,mL+N_p+l+1} = h_{k,mL+N_p+1} \tag{29}$$

$$\text{for } \begin{cases} m \in \{0, \ldots, M-1\} \text{ and } l \in \{1, \ldots, L-1\}, \\ m = M \text{ and } l \in \{1, \ldots, l_{max}\}, \end{cases}$$

where $l_{max} \equiv N_s - 1 \pmod{L}$.

This principle can of course be applied in the frequency domain. However, experiments showed that performance was unsatisfactory, due to the short coherence bandwidth. Note that other interpolation methods exist and could have be chosen in the design of the algorithm. For instance, linear interpolation also has very small computational complexity and thus would be envisagable. However, since piecewise interpolation has the lowest complexity of all interpolation methods and shows good performance, we selected it over other possible interpolation methods.

C. Discussion

In the case of vehicular communications, transmitters and receivers need to communicate in many different environments [1], in situations where only one unit is moving (V2I) or both (V2V), and with different rates available in the standard. As a consequence, it is critical that receivers show enough flexibility to be able to adapt to various situations. Our approach was to rely the least on channel models, and rather let the receiver estimate most of the channel's properties based on measurements. As one model cannot describe accurately all the situations that are encountered, we obtain a more versatile receiver than the model-based ones. Moreover, we proposed a scalable algorithm that allows adapting the computational complexity. Performance will vary depending on the environment, and it is beneficial to use the low complexity algorithm if the channel is favorable, and thus computational power is not wasted.

IV. Complexity Analysis

One of our objectives is to design a receiver that is suitable for hardware implementation. In this section, we discuss complexity aspects of our algorithms, and compare them with the design proposed in [4]. On top of playing a crucial role in the performance of receivers, channel estimation is also the most time-consuming part of the algorithms. We therefore focus on the part which has the biggest importance and differs most between the different decoder proposals.

In this section, we will give estimates of the complexity using Landau notation [36]. Instead of providing separate measures for the different factors influencing the total complexity, or taking only the one with the biggest power or the biggest

value, we present all these factors together. We believe that this approach gives a more accurate overview of the complexity while dealing with equivalents, without being as thorough as the count of FLOPS (FLoating point Operations Per Second).

The algorithm proposed in [4] uses a reduced-rank Wiener filter for channel estimation. In [4, Eq. 57], a formula estimating the number of FLOPS is provided. It is found to depend on the number of OFDM symbols N_s, the number of subcarriers N_{sc} and the dimension of the subspace matrix D. This figure essentially accounts for the inversions of non-diagonal matrices in the Wiener filtering operation. The values of D given in [13] and [4], namely 75 for 72 symbols and 38 for 38 symbols, suggest a linear relation with the frame length, which our simulations subsequently confirmed. Clearly, although the reduced dimension represents an interesting improvement on the full rank filter, this approach still leads to a high complexity and would therefore be unsuitable for long frames as well as hardware implementation. Indeed, the filter has to be computed for each iteration. The rank reduction also requires a subspace selection to choose the adequate matrix rank for each received frame. When complying to the standard, which includes no postamble, the selection only depends on the evaluated Doppler power Spectral Density (DSD) support since maximum Power Delay Profile (PDP) support has to be assumed. This evaluation is performed on the pilot subcarrier symbols, once per frame. Its complexity can be seen as cubic with respect to the number of symbols. It should be noted that without postamble, the gains in complexity are smaller due to the required choice of the highest rank for the PDP hypothesis matrices.

In this paper, we propose a lower complexity design based on the factor graph framework. The channel estimation was described in Fig. 2. We remind that one channel coefficient node is needed for each symbol we want to estimate; considering that only N_{scu} subcarriers are used, the number of channel coefficients nodes amounts to $N_s N_{scu}$. We consider the adaptation (or decimation) factor L as well. In the equalizer, the number of operations to carry out is proportional to the number of channel coefficient to estimate, which is equal to the total number of channel coefficients divided by the L factor (we estimate one every L coefficient); in this approach, the cost of interpolation was not taken into account, which seems reasonable since we used constant piecewise interpolation. At each iteration in the graph, channel coefficient nodes will perform the sum-product algorithm on Gaussian distributed random variables, which requires two parameters, namely mean and variance, to be entirely defined. These distributions come from two neighboring frequency transfer nodes and two time transfer nodes, as well as the corresponding observation node. The total number of computations required will be linear with respect to the number of OFDM symbols considered and the number of subcarriers. This represents indeed a major improvement over designs such as the one proposed in [4].

The final results are summarized in Table II, where the number of data subcarriers is denoted as N_{scd}. For completeness, we also provide complexities of two other non iterative algorithms proposed for IEEE 802.11p: [10] and [7], as well as the more conventional Least Squares (LS) and comb LS techniques presented in [5]. Our design has the lowest complexity among the iterative algorithms presented.

TABLE II
COMPLEXITY COMPARISON OF CHANNEL ESTIMATION ALGORITHMS

Algorithm	**Complexity in $\mathcal{O}$**	**Source**
LS	N_{scd}	
CLS-linear	$N_s N_{scu}$	[5]
Zhao et al.	$N_s N_{scd}$	[10]
Fernandez et al.	$N_s N_{scu}$	[7]
Zemen et al.	$N_s^3 N_{sc} N_{it}$	[4]
Scalable FG	$N_s N_{scu} N_{it}/L$	

For hardware implementations, our channel estimation algorithm has the advantage of requiring mainly simple algebraic operations, that is, additions and multiplications. This is an important feature, since latency can be a relevant metric for receiver designs. However this performance metric depends heavily on the implementation, and therefore we chose to not investigate rigorously the latency performance of our algorithm, as this would require a comparison of explicit implementations which is outside of the scope of this work. Note that, while our algorithm still requires the computation of divisions, exponentials and logarithms, higher complexity computations, such as the matrix inversions required by [4] or function integrations, are not needed. Furthermore, we should highlight that it is possible to parallelize some parts of the algorithm, such as information propagation in the frequency and time dimensions of the graph, by taking advantage of the scheduling. This makes our algorithm very suitable for hardware implementation.

V. NUMERICAL RESULTS

In this section we present the numerical results obtained while simulating data transmission using an IEEE 802.11p compliant transmitter and the receiver described in Section III. First, we display the performance of the algorithm in terms of error rate. We show the influence of various parameters such as the frame length or the speed of the vehicles on the performance. We then show measurements for the complexity in terms of execution times. Finally, we compare different receivers both in terms of performance and complexity, allowing us to discuss the complexity performance trade-off.

A. BER and FER Performance

We perform Monte Carlo simulations, with up to 2000 realizations, to evaluate the performance of the receiver in terms of BER and FER for various values of normalized SNR, noted E_b/N_0 and expressed in dB. We transmit data at the default rate of 6 Mbps, which implies employing QPSK modulation and no puncturing. The channel is simulated by a tapped delay line with an exponentially decaying average power delay profile. More precisely, we considered a channel with 20 taps which are spaced by $T_s = 100$ns; a power delay profile where the mean value exponentially decays (RMS delay spread is $4T_s$). Each tap is Rayleigh distributed and the variations correspond to a Jakes spectrum model with Doppler

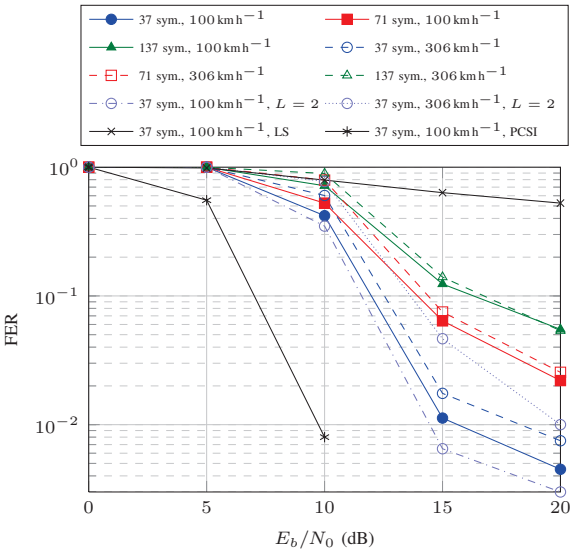


Fig. 8. FER as a function of E_b/N_0 for frame lengths $N_s \in \{37, 71, 137\}$ and relative speed $v \in \{100, 306\}$ (km h^{-1}).

frequency $f_d = f_c v/c$ where the carrier frequency is given by $f_c = 5.9$GHz. Parameters are chosen such that $\tau_{RMS} = 400$ ns and the support is 1.9 µs. Finally, in this Section V-A, we assume that the interpolation factor $L = 1$ is chosen. The impact of the interpolation factor and a thorough analysis of its influence on the achievable performance are illustrated later in this Section in Figures 13 and 14.

TABLE III
NORMALIZED DOPPLER FREQUENCY FOR GIVEN SPEED AND FRAME LENGTH.

v ($km.h^{-1}$)	N_{ofdm} 37	71	137
100	0.162	0.310	0.599
306	0.495	0.950	1.83

Fig. 8 shows the FER as a function of E_b/N_0, for frames containing 37, 71, and 137 symbols (excluding pilots), which corresponds to payloads of 200, 400, and 800 bytes respectively, and relative speeds between transmitter and receiver of 100 km h^{-1} and 306 km h^{-1}. We also represent in black curves in the figure the performance of least squares (LS) estimation and the scenario with perfect channel state information (PCSI in the legend) for frames of 37 symbols and a relative speed of 100 km h^{-1}. Furthermore, we represent in light blue dotted lines the FER performance for 37 symbols and the adaptation factor $L = 2$, which will be discussed later in Section V-C. We also provide the equivalent normalized Doppler frequencies corresponding to given frame length and relative speed in Table III. We verify that the case with perfect channel estimates acts as an upper bound to the FER performance of our algorithm, while the simple LS estimation algorithm is outmatched by our algorithm. As expected, we observe that the performance is affected by the frame length and the speed, and in particular that the FER increases with the speed and the number of symbols per frame. Short frames usually have a duration shorter than the coherence time and are not subject

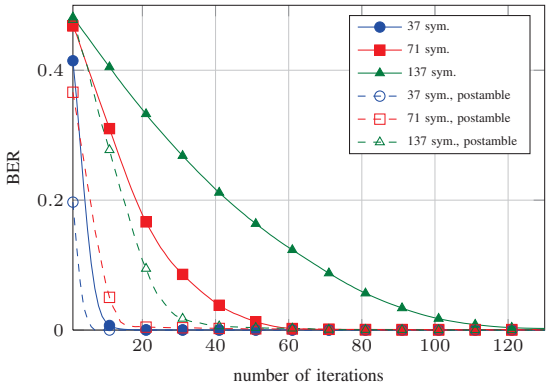


Fig. 9. Evolution of the BER over the iterations for frame length $N_s \in \{37, 71, 137\}$, with and without an additional postamble.

to large variations. On the other hand, the duration of the medium frames is comparable to the coherence time and thus they are affected by non negligible variations. The extent of the phenomenon is even bigger in the case of large frames. The coherence time is related to the relative speed and at higher velocities the effect of time variations becomes worse. It is also important to note that the longer the frame is, the larger the probability to encounter deep fading is. Deep fading typically occur several times during a long frame and impairs strongly the channel estimation and thus the decoding of the frames. When comparing these results with those obtained when reproducing the algorithm proposed in [13], we observe that our algorithm is more robust to an increase in frame length. Transmitting long frames reliably ultimately increases the payload; thus it is an important factor to monitor. There exists some debate about the FER for which communication is considered to be reliable. Since this obviously depends on the quality of service requirements, we follow [3] where an FER smaller than 0.1 is the target. This value is reached in all case, at different SNRs.

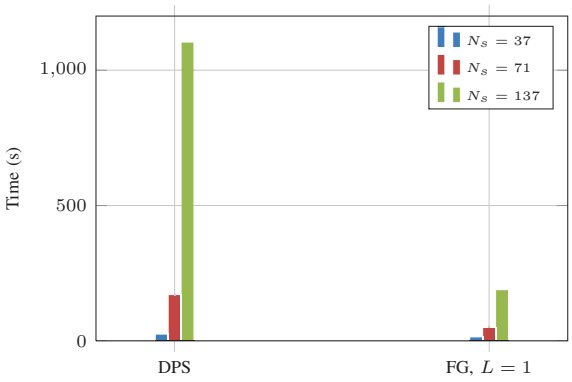
Another important measure when dealing with iterative receivers is the speed of convergence. Fig. 9 shows the evolution of the BER over the iterations for three frame lengths. We also show the influence of placing a postamble, originally introduced in [13]. The speed of convergence decreases when the frame length increases. Convergence is fast for short frames, as only 10 iterations are required to reach the error floor. However, medium and long frames require more iterations, namely 70 and 130 respectively. Indeed ,if the frame length increases (i.e. the number of transfer nodes increases), more iterations will be required to get the same accuracy in terms of LLR compared to shorter frames as explained in Section III-B. Since the decision rule for decoding is based on the LLRs, more decoding errors occur when the LLRs carry insufficient information and therefore more iterations are needed to obtain the same BER performance. This justification explains why placing a postamble reduces almost by half these numbers, since it effectively decreases the maximum distance from a pilot symbol to a transfer node. We also observe the gain obtained by using an iterative structure. At the first iteration, only the information from the pilots is used, and the

channel estimation is unsatisfactory, leading to high BERs. The iterative structure allows to reach much lower error rates.

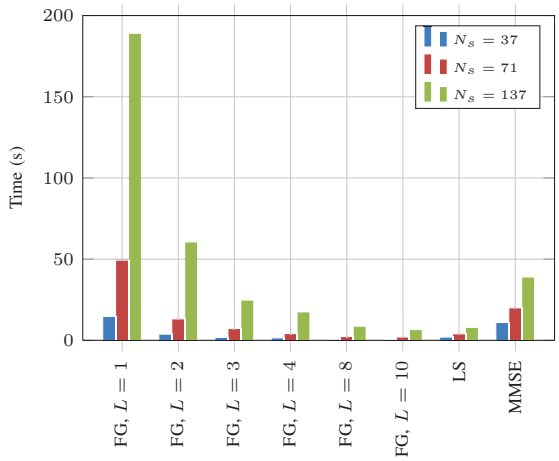
B. Complexity

The measure of complexity is given here as the time of execution required to decode one frame. As opposed to the theoretical complexities presented in Section IV, execution times take into account all the components of the algorithm. Note that execution times were measured on the MATLAB link level simulation and may vary based on the load at the time of the measure. Simulations were performed on the same computer with 2.5 GHz Intel Core i5 processor, with 4 GB of 1600 MHz DDR3 memory, and MATLAB software release 2013. The values do not represent what would happen in the case in an actual hardware implementation. The previous rigorous analytical discussion on the complexity looking at the complexity order using the Landau notation might not be informative enough for the complexity assessment of the algorithms and therefore, we complement the discussion by measurements of the execution time. While measurement of the execution time can be seen as a less rigorous performance metric of the complexity, it is however a very good indicator of the computational complexity of the different algorithms since the simulations are performed under the same simulation conditions. Thus we will focus more on the comparison of different receivers rather than the absolute durations. Fig. 10 presents the execution times of various receivers. We compare our algorithm for several factors of decimation, with the design presented in [4], an iterative Minimum Mean Square Error (MMSE) receiver, and a more classical non iterative receiver with channel estimation based on LS. Note that there exist several advanced variations of the LS and MMSE algorithm (see, e.g., [37]). However, the modified LS method in this reference comes with an increased complexity while the potential complexity reduction of the modified MMSE method comes with a worse performance. Therefore, we believe that using the standard MMSE and LS methods as relevant benchmark curves is justified. The execution times necessary to decode frames of various lengths are displayed and expressed in seconds. We observe in Fig. 10a that the gain in complexity compared to the algorithm proposed in [4] is considerable, especially with longer frames. This is consistent with the dependence of the complexity on N_s found in Section IV. Furthermore, it is of particular interest to analyze the influence of L on the complexity and the execution times of our algorithm. Without decimation ($L = 1$), we observe on Fig. 10b that our algorithm takes more time than the LS and MMSE receivers. However, by increasing L, the execution times decrease and our algorithm becomes faster.

Note that the increase in execution time with the frame length does not seem linear in the case of our algorithm, as opposed to what was explained in Section IV. This is due to the fact that the number of iterations required to converge is higher for long frames than short frames, as shown in Fig. 9. However, its linear dependency in N_s and relative simplicity regarding the computations largely compensates for that, as it is still much faster than [4] without decimation, and faster than the LS and MMSE receivers with $L \geq 8$.



(a) Comparison with [4] (DPS)



(b) Comparison with MMSE and LS, for several values of L

Fig. 10. Execution time of our algorithm (FG) for $L \in \{1, 2, 3, 4, 8, 10\}$, the one proposed in [4], denoted as DPS, an MMSE receiver and an LS receiver and for $N_s \in \{37, 71, 137\}$.

C. Complexity Performance Trade-off

In this section, the number of data symbols in the frame is $N_s = 37$ and the relative speed between vehicles is $v = 100\,\mathrm{km\,h^{-1}}$.

Fig. 11 depicts the complexity performance trade-off achieved by different receivers in terms of BER for $E_b/N_0 = 15\,\mathrm{dB}$. We compare our algorithm for several values of the adaptation factor L to other algorithms from the literature. Whether a trade-off is acceptable or not must be decided based on the application, but we suggest a classification of receivers. Better receivers are closer to the origin. The further the point is from the origin, the least satisfactory the trade-off is. For $L \in \{2, 3, 4, 8\}$, our algorithm provides reliable communications (we consider a communication reliable if BER ≤ 0.01), for a low complexity. However, for $L = 10$, the performance is too low. This is also the case for the LS receiver. Our algorithm without decimation shows good performance, but the complexity is quite high, leading to an unsatisfactory trade-off. Indeed, the BER value is above the one achieved when $L = 2$ while the complexity is higher. It is quite interesting to note that $L = 2$ performs better than $L = 1$, which might seem counter-intuitive. While we do not have an analytical justification for this behavior due to the complicated frame structure, our conjecture is as follows. First,

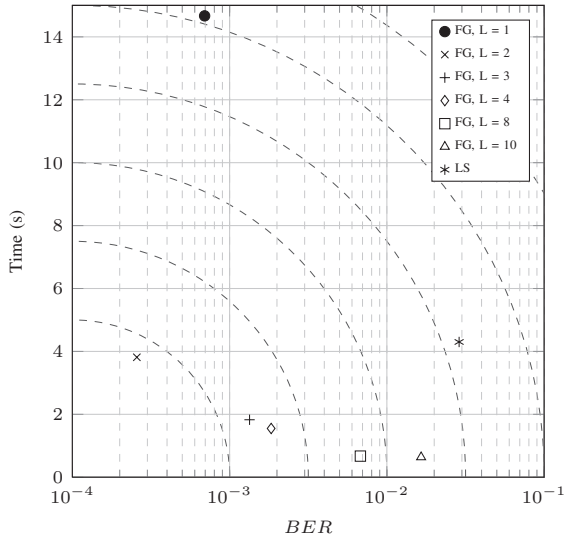


Fig. 11. Complexity Performance trade-off of different receivers in the case where $E_b/N_0 = 15\,\text{dB}$. The performance is measured as the BER and the complexity as the execution time in seconds.

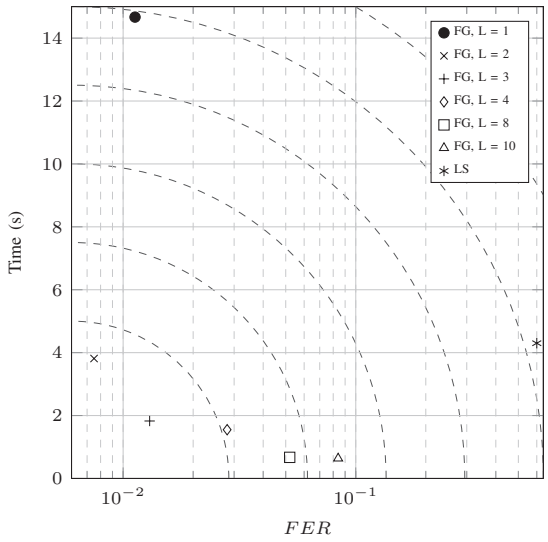


Fig. 12. Complexity Performance trade-off of different receivers in the case where $E_b/N_0 = 15\,\text{dB}$. The performance is measured as the FER and the complexity as the execution time in seconds.

since the SPA works better on smaller graphs, the reduction of the graph from $L = 1$ to $L = 2$ might lead to an improvement of the performance of the algorithm. For values of L larger than 2, the decrease in the quality of the estimation due the higher variations between two consecutive nodes becomes too important. Moreover, since the variances of the estimators increase at each transfer node, the variance of the estimation is overall decreased when considering a smaller graph, i.e. when $L = 2$ compared to $L = 1$, which could possibly result in the better performance.

In Fig. 8, the performance in terms of FER is shown for $L = 2$ in comparison to $L = 1$. It is noteworthy that while $L = 2$ is optimal for a frame lenght of 37 symbols and a speed of $100\,\text{km}\,\text{h}^{-1}$, it is not true for all speeds. Indeed, from

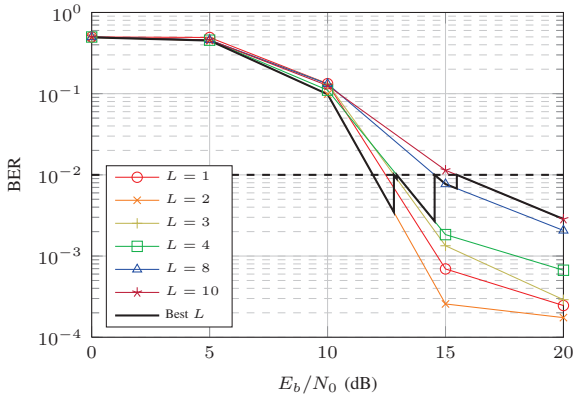


Fig. 13. BER as a function of E_b/N_0 for various adaptation factors.

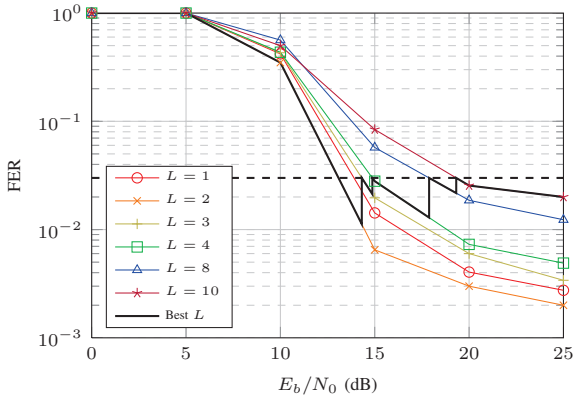


Fig. 14. FER as a function of E_b/N_0 for various adaptation factors.

Fig. 8, we can see that for a speed of $306\,\text{km}\,\text{h}^{-1}$, $L = 2$ performs worse than $L = 1$. Note that decimating the graph when the relative speed of the communication node is high, strongly increases the channel variations between the nodes of the decimated graph. This highlights the importance of the system parameters for the choice of the adaptation factors in our scalable algorithm.

In Fig. 12 we illustrate the complexity performance trade-off achieved by different receivers in terms of FER for $E_b/N_0 = 15\,\text{dB}$. Similar to Fig. 11 we compare our algorithm for several values of the adaptation factor L to other algorithms from the literature. In particular we observe that we can draw the same conclusions as for the BER curves analysis. However if we consider a benchmark of FER ≤ 0.03, which is more restrictive than the generally assumed 0.1 threshold (for which the $L = 8$ performance would be acceptable), we notice that for $L = 8, 10$, the performance is too low.

As seen in Fig. 11, our algorithm can offer an interesting complexity performance trade-off. Moreover, by adapting the adaptation factor, this trade-off can be kept to the most advantageous value when the SNR varies. Note that the interpolation factor L should in general be adapted according to the channel conditions and the sole knowledge of the channel SNR does not guarantee the optimality of a given L for all channel models. Fig. 13 shows the performance in terms of BER

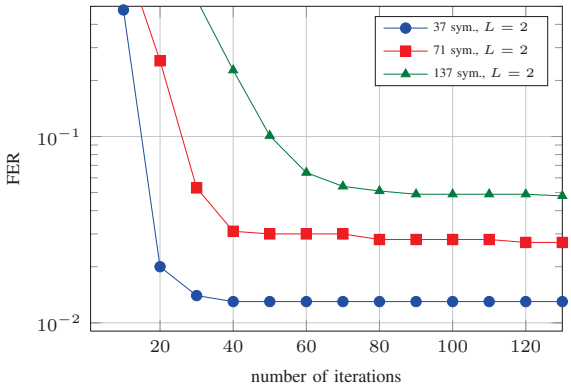


Fig. 15. Evolution of the FER over the iterations for frame length $N_s \in \{37, 71, 137\}$ and $L = 2$

of the receiver for different decimation factors and suggests a selection of the aforementioned factor based on the SNR value. A threshold for the BER is fixed, and the largest L that guarantees a BER less than the threshold, at a certain SNR and for given channel conditions, is chosen. In Fig. 13, we choose a BER of 0.01 as the threshold and obtain a plot of the BER as a function of the SNR with the best factor L. For example, a receiver communicating over a channel where $E_b/N_0 = 10\,\text{dB}$ will use $L = 2$. If E_b/N_0 increases to $15\,\text{dB}$, using $L = 8$ will still guarantee reliable communications, as the BER is below the threshold, but at a lower complexity. The receiver will thus adapt and switch to that value of L. Similarly we depict in Fig. 14 a plot of the FER as a function of the SNR with the best factor L, given a threshold of 0.03 for the FER. We observe a similar behavior in the choice of the interpolation factor, which allows the system to minimize the complexity while guaranteeing acceptable performance.

Finally in Fig. 15, we depict the evolution of the FER over the iterations for three frame lengths for the adaptation factor $L = 2$, which was shown to be optimal for a frame length of 37 symbols. We observe that the choice of $L = 2$ leads to a faster convergence than for $L = 1$, by comparing the curves with the ones in Fig. 9. This shows another advantage of using the scalable algorithm over the non-adaptive one, as the performance limit is achieved in a shorter time.

VI. Conclusion

Obtaining reliable systems in the case of vehicular communications is challenging, especially regarding channel estimation. Receivers face fast varying channels and deep fading. Iterative receivers allow estimating the channels despite the variations, but they introduce an increase in the computational complexity. In this paper we proposed a scalable algorithm, which can be adapted to the environment by choosing the best available trade-off between complexity and performance. Numerical results showed that our algorithm achieves reliable communications, characterized by low BERs and FERs, at medium and high SNR. This is a clear improvement compared to non-iterative receivers. While state-of-the-art high complexity iterative receivers perform better at low SNR, the algorithm proposed in this article surpasses them at high SNR. Moreover, the gain in complexity is considerable, especially with longer frames, in which case the execution time is divided by a factor up to 100. Combining the performance and the complexity showed that we reach a satisfying trade-off with this design.

References

[1] C. F. Mecklenbräuker, A. F. Molisch, J. Karedal, F. Tufvesson, A. Paier, L. Bernadó, T. Zemen, O. Klemp, and N. Czink, "Vehicular channel characterization and its implications for wireless system design and performance," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1189–1212, Jul. 2011.

[2] F. Knorr, D. Baselt, M. Schreckenberg, and M. Mauve, "Reducing traffic jams via VANETs," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 8, pp. 3490–3498, Oct. 2012.

[3] P. Alexander, D. Haley, and A. Grant, "Cooperative intelligent transport systems: 5.9-GHz field trials," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1213–1235, Jul 2011.

[4] T. Zemen and A. Molisch, "Adaptive reduced-rank estimation of non-stationary time-variant channels using subspace selection," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 9, pp. 4042–4056, Nov. 2012.

[5] L. Bernadó, N. Czink, T. Zemen, and P. Belanović, "Physical layer simulation results for IEEE 802.11p using vehicular non-stationary channel model," in *IEEE International Conference on Communications (ICC), Cape Town, South Africa*, May 2010.

[6] K.-P. Chou and C.-K. Sun, "A modified decision directed channel tracking of 802.11p OFDM system," in *11th International Conference on ITS Telecommunications (ITST), Saint-Petersburg, Russia*, Aug. 2011.

[7] J. A. Fernandez, D. D. Stancil, and F. Bai, "Dynamic channel equalization for IEEE 802.11p waveforms in the vehicle-to-vehicle channel," in *48th Annual Allerton Conference on Communication, Control, and Computing, Illinois, U.S.A.*, Sep. 2010.

[8] C.-S. Lin, C.-K. Sun, J.-C. Lin, and B.-C. Chen, "Performance evaluations of channel estimations in IEEE 802.11p environments," in *International Conference on Ultra Modern Telecommunications Workshops (ICUMT), Saint-Petersburg, Russia*, Oct. 2009.

[9] C.-S. Lin and J.-C. Lin, "Novel channel estimation techniques in IEEE 802.11p environments," in *IEEE 71st Vehicular Technology Conference (VTC Spring), Taipei, Taiwan*, May 2010.

[10] Z. Zhao, M. Wen, X. Cheng, C.-X. Wang, and B. Jiao, "A novel effective channel estimation scheme applicable to IEEE 802.11p," in *12th International Conference on ITS Telecommunications (ITST), Taipei, Taiwan*, Nov. 2012.

[11] R. Budde and R. Kays, "Challenges and improvement strategies for the physical layer in vehicular communications," in *18th European Wireless Conference (EW), Poznan, Poland*, Apr. 2012.

[12] W. Cho, S. I. Kim, H.-K. Choi, H. seo Oh, and D.-Y. Kwak, "Performance evaluation of V2V/V2I communications: The effect of midamble insertion," in *1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace Electronic (VITAE), Aalborg, Denmark*, May 2009.

[13] T. Zemen, L. Bernadó, N. Czink, and A. F. Molisch, "Iterative time-variant channel estimation for 802.11p using generalized discrete prolate spheroidal sequences," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 3, pp. 1222–1233, Mar. 2012.

[14] M. Kang, S. Zhang, and X. Lin, "An approach to combat frequency selective fading in vehicular communications using SK-OFDM," in *International Conference on ICT Convergence (ICTC), Jeju Island, South Korea*, Oct. 2012.

[15] G. Kiokes, A. Amditis, and N. K. Uzunoglu, "Simulation-based performance analysis and improvement of OFDM - 802.11p system for vehicular communications," *IET Intelligent Transport Systems*, vol. 3, no. 4, pp. 429–436, Dec. 2009.

[16] M. C. Valenti and B. D. Woerner, "Iterative channel estimation and decoding of pilot symbol assisted turbo codes over flat-fading channels," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 9, pp. 1697–1705, Sep. 2001.

[17] A. Dapena, T. M. Fernández-Caramés, J. A. Garcia-Naya, and L. Castedo, "A decision-aided channel estimation strategy for the IEEE 802.11p standard," in *3rd International Workshop on Cognitive Information Processing (CIP), Baiona, Spain*, May 2012.

[18] M. Siti, A. Assalini, E. Dall'Anese, and S. Pupolin, "Low complexity decision-directed channel estimation based on a reliable-symbol selection strategy for OFDM systems," in *IEEE International Conference on Communications Workshops (ICC), Cape Town, South Africa*, May 2010.

[19] A. P. Worthen and W. E. Stark, "Unified design of iterative receivers using factor graphs," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 843–849, Feb. 2001.

[20] Z. Shi, T. Wo, P. A. Hoeher, and G. Auer, "Graph-based soft iterative receiver for higher-order modulation," in *12th IEEE International Conference on Communication Technology (ICCT), Nanjing, China*, Nov. 2010.

[21] T. Wo, C. Liu, and P. A. Hoeher, "Graph-based soft channel and data estimation for MIMO systems with asymmetric LDPC codes," in *IEEE International Conference on Communications (ICC), Beijing, China*, May 2008.

[22] C. Knievel, P. A. Hoeher, A. Tyrrell, and G. Auer, "Multi-dimensional graph-based soft iterative receiver for MIMO-OFDM," *IEEE Transactions on Communications*, vol. 60, no. 6, pp. 1599–1609, Jun. 2012.

[23] ——, "Improving Multi-Dimensional Graph-Based Soft Channel Estimation," in *IEEE 75th Vehicular Technology Conference (VTC Spring), Yokohama, Japan*, May 2012.

[24] S. Marsili, "DC offset estimation in OFDM based WLAN application," in *Global Telecommunications Conference (GLOBECOM), Dallas, Texas, U.S.A.*, Nov. 2004.

[25] E. G. Ström, "On medium access and physical layer standards for cooperative intelligent transport systems in Europe," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1183–1188, Jul. 2011.

[26] A. Sassi, F. Charfi, L. Kamoun, Y. Elhillali, and A. Rivenq, "OFDM transmission performance evaluation in V2X communication," *International Journal of Computer Science Issues*, vol. 9, no. 3, pp. 141–148, Mar. 2012.

[27] G. Acosta-Marum and M. A. Ingram, "Six time- and frequency- selective empirical channel models for vehicular wireless LANs," *IEEE Vehicular Technology Magazine*, vol. 2, no. 4, pp. 4–11, Dec. 2007.

[28] U. Madhow, *Fundamentals of Digital Communication*. Cambridge University Press, 2008.

[29] G. Tauböck, F. Hlawatsch, D. Eiwen, and H. Rauhut, "Compressive estimation of doubly selective channels in multicarrier systems: Leakage effects and sparsity-enhancing processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 255–271, April 2010.

[30] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[31] T. Wo, C. Liu, and P. A. Hoeher, "Graph-based iterative Gaussian detection with soft channel estimation for MIMO systems," in *7th International ITG Conference on Source and Channel Coding (SCC), Ulm, Germany*, 2008, pp. 1–6.

[32] L. Bernadó, T. Zemen, F. Tufvesson, A. F. Molisch, and C. F. a. Mecklenbräuker, "The (in-) validity of the WSSUS assumption in vehicular radio channels," in *23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), Sydney, Australia*, Sept. 2012.

[33] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (corresp.)," *IEEE Transactions on Information Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[34] S. J. Johnson, *Iterative Error Correction: Turbo, Low-Density Parity-Check and Repeat-Accumulate Codes*. Cambridge University Press, 2009.

[35] H. Wymeersch, *Iterative Receiver Design*. Cambridge University Press, 2007.

[36] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. MIT Press, 2001.

[37] J. J. van de Beek, O. Edfors, M. Sandell, S. K. Wilson, and P. O. Börjesson, "On channel estimation in OFDM systems," in *IEEE Vehicular Technology Conference, Chicago, U.S.A*, 1995.

Olivier Goubet received the M.Sc. degree from Supélec, France and from the Royal Institute of Technology (KTH), Stockholm, both in 2013. From March 2013 to May 2013, he was with the Communication Theory Division at KTH, as a research engineer. In June 2013, he joined the signal processing department at Thales Communications and Security. His research interests include vehicular communications and signal processing for digital communications, with emphasis on equalization, channel estimation and iterative and turbo algorithms.

Gwilherm Baudic received the M.Sc. degree from the Institut Supérieur de l'Aéronautique et de l'Espace, Toulouse, France (ingénieur ISAE-ENSICA) in 2013. He spent his final year at the Royal Institute of Technology (KTH), Stockholm, Sweden, with his Master Thesis conducted at the Communication Theory Lab. Since October 2013, he has been working as a Ph.D. student in the Computer Science department (DMIA) of the Institut Supérieur de l'Aéronautique et de l'Espace. His research interests include delay tolerant networks and vehicular communications.

Frédéric Gabry (S'10) received the M.Sc. degree from Telecom Paristech, France (ingénieur Telecom ParisTech) in 2009 and the M.Sc degree and Lic. Eng. degree from the Royal Institute of Technology (KTH), in 2009 and 2012, respectively. Since October 2009, he is working towards a Ph.D. degree in telecommunications in the Communication Theory Division at KTH. His research interests include physical layer security, game theory and cooperative communications.

Tobias J. Oechtering (S'01-M'08-SM'12) received his Dipl-Ing degree in Electrical Engineering and Information Technology in 2002 from RWTH Aachen University, Germany, his Dr-Ing degree in Electrical Engineering in 2007 from the Technische Universität Berlin, Germany, and his Docent degree in Communication Theory in 2012 from KTH Royal Institute of Technology. Between 2002 and 2008 he has been with Technische Universitt Berlin and Fraunhofer German-Sino Lab for Mobile Communications, Berlin, Germany. In 2008 he joined the Communication Theory Lab at the KTH Royal Institute of Technology, Stockholm, Sweden and has been an Associate Professor since May 2013. Presently, he is serving as an editor for IEEE Communications Letters. Dr. Oechtering received the "Förderpreis 2009" from the Vodafone Foundation. His research interests include vehicular communication, communication and information theory, physical layer security, signal processing for wireless communication, as well as communication for networked control.